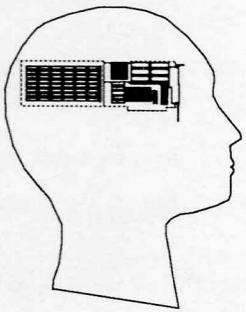
RM User

Vol 3

No 1 *July 1988*



Thinking about the conference?

The Journal of the RM User Group

HELP-LINE

The User Group contains a great body of expertise amongst its members and the help-line exists to act as a channel for spreading knowledge and especially answers to your problems.

Remember, your problems are also our problems!

If you are in difficulty, please feel free to ring any of the following numbers. We may not, individually, know the answers, but we hope that we may know someone who knows someone who.....!

Robin Davies:	0902-310360 (home) 0902-305958 (work)
Ken Maynard:	0438-354809 (home) 0438-352467 (work)
Les Watson:	0452-741203 (home) 0242-513836 (work)
Tim Davey-Winter	0865-278826 (work) 0865-244645 (home)
Paul Maskens	01-464-7539 (work)

Anybody else on the committee (see inside back cover) can be contacted if none of the above can help or is available.

Remember, the Group is on **TELECOM GOLD 72:MAG 90380**

Advertising in the Journal

Full- and half-page advertisements can be placed in the Journal and the rates are as follows:

Full Page: £50 Half Page: £30

These prices assume black & white camera-ready artwork. Any typesetting is charged at cost.

The deadline for the next issue is September 9th

 © RM User Group 1988

Index

Editorial 4

Letter 5

Library News 7

RM News 9

Review: Clipper - a dBase III+ Compiler 11

dBase III+ & II in the Classroom 15

A Music Database in dBase 17

Authoring Shells 34

Writing for the Journal 37

Communications with Nimbus RM Basic 39

480Z Quest to dBase File Transfer 44

Advanced Networking 46

Micro Mart 58

Executive Committee 59

Advertisers' Index

Peter Bishop Associates 6
Aurora Educational Software 14
Whittle Systems 8, 30,31
ITEM 38
Primrose Publishing Back Cover

 © RM User Group 1988

Letters

Dear Network managers, (care of the editor RM USER),

Monitoring users disk space on X series stations.

Those of you using AX and VX stations will no doubt have read the Network release 2.1 Managers handbook extension. It says on page 24,

"The disk space monitoring program NETDSK does not work in IBM mode. There is no point in offering it as a logon program nor in setting the 'Disk space' in the main details screen".

It does not however point out that if you do set the 'Disk space' in the main details screen, you can use NETDSK by changing to RM mode.

I am now doing this successfully by using the following series of logon programs for X series users :

Logon program 1 Q:\MODE.EXE RM
Logon program 2 P:\NETDSK.EXE
Logon program 3 Q:\MODE IBM

Similarly I am using the same sequence of programs at log off, but with the /SOFT option on NETDSK.

Yours Sincerely,

Charles C. Marshall

Teesside Polytechnic,
School of Information Engineering,
(Computer and Maths Sciences building),
Borough Road,
Middlesbrough
Cleveland
TS1 3BA

Telephone work : (0642) 218121 Extn 4186
home (0642) 231804

 © RM User Group 1988

Software ... from Peter Bishop Associates

AlphaStat £49.00 + VAT

A mathematical, statistical and graphics software pack.

AlphaCalc £39.00 + VAT

A versatile spradsheet which combines ease of use with the power of a full commercial spreadsheet.

AlphaCalc Activity Pack £20.00 + VAT

A set of worksheets and spreadsheets on disk for use with AlphaCalc. A wide range of activities is covered.

DeltaStat – New Version £95.00 + VAT

A powerful mathematical, statistical and graphical software pack for the analysis of research data.

AlphaPlot £29.00 + VAT

Plots line graphs, a variety of bar graphs and pie charts from AlphaStat and AlphaCalc data.

DeltaPlot £59.00 + VAT

As AlphaPlot, also reads DeltaStat data.

Multiplan to AlphaCalc transfer utility £15.00 + VAT

dBaseII to DeltaStat transfer utility £45.00 + VAT

Computers supported:

AlphaStat, AlphaCalc & AlphaPlot : RM 380Z, 480Z and Nimbus range.

DeltaStat & DeltaPlot: RM Nimbus range.

The same versions run stand-alone and on networks.

Peter Bishop Associates
23 Larkfield Road
Richmond
Surrey TW9 2PG
Telephone: 01-948 4382

PBA

 © RM User Group 1988

Nimbus Software Library

Once upon a time life was pleasant and relatively easy. My pupils enjoyed working with SBASAG and INTBASAG on the 480Z machines. Then came the Nimbus PC-186. We decided the 6th Form pupils should be working with the new machines as soon as possible.

First we used RM Basic.....enough said? All those discrepancies between the manual and what was actually possible with the software!

Next there was a need to introduce a more professional style of programming to the pupils, plus considerable encouragement from our local department of ICL to teach them Pascal. I had investigated RM Pascal for the Nimbus PC-186 and, to my amazement, could find no way of saving a compiled program to disk! Has anyone found how to do this? I cannot believe that I have not missed some vital instruction in the manual. How was I supposed to demonstrate one of the main benefits of using a compiler - that of finishing up with a compiled program which would run independently? At that stage, I had almost decided to move their lessons back into the room with the 480Z network.

Then I thought of Turbo Pascal. I had to reject Version 4 as my machines do not have sufficient memory to run the IBM emulator they need as well as Turbo. Therefore it was V.3 or nothing. The next problem? You guessed it! The 6th Form, brought up on SPA software with all of its graphics handling routines, turned up their noses and said:

Wot, no graphics?

Although I declared there was a lot of work to be done which does not require graphics, I had to admit at the same time that I do spend a lot of time criticising their methods of displaying data and recommending ways of conveying the true implications of their results.

At last I discovered the Turbo Graphics package on Volume 1 of the User Group Nimbus Software Library. Now we have the graphics facilities we know so well **and** Pascal **and** compiled programs on disk! We can no do line and marker drawing, area filling, text plotting, block moves, sprite handling, mouse/joystick control and ..... I can recommend it.

Many thanks to those of you who responded with ideas for the development of the Nimbus Library. I am following up each one and will reply to everyone as soon as possible. Ideas are always welcome. Are there any programs written by members who would not mind sharing them with the User Group?

Each Nimbus Library Volume costs £5. There is an extra charge of £2.50 if you do not send a formatted 3.5" disk. Please make cheques payable to RMUG.

 © RM User Group 1988

Price List — Education June 1988

Whittle Systems

For
YOUR NIMBUS PC
Expansion

APPROVED
CUMANA DISTRIBUTOR

HARDWARE

CSR200	5.25" DSDD external floppy drive	**£124.00**

EXTERNAL WINCHESTERS

CXM10	10 Megabyte exchangeable cartridge	**£499.00**
CRM30	30 Megabyte	**£555.00**
CRM40H	40 Megabyte, high speed	**£850.00**
CRM60H	60 Megabyte, high speed	**£935.00**
CXM10+30	10 Megabyte exchangeable + 30 Megabyte fixed	**£999.00**
	Surcharge for 2nd/3rd External	**£20.00**

CABLES

DTI modem cable	**£9.95**
Serial Printer cable	**£9.95**
Low cost parallel port printer cable	**£9.95**
8 bit parallel printer cable	**£19.95**
Internal & external Winchester adaptor cable	**£25.00**
Internal & external Floppy adaptor cable	**£29.95**

MEDIA

10 Mb cartridges for CXM10	each	**£25.00**
	box of 3	**£66.00**

SOFTWARE

NIMBUS VIEWDATA III Complete local Viewdata package	**£125.00**
NIMBUS VIEWDATA II to **NIMBUS VIEWDATA III** upgrade	**£30.00**
TRAPDOOR The ultimate debugging aid *(also available for IBM PC, APRICOT, PC, XI, XEN)*	**£49.95**
ZEM Z80 CP/M Emulator	**£49.95**

Network prices on application

All prices exclude P & P and Vat.

53, Besselsleigh Road,
Wootton,
Abingdon, Oxford OX13 6DX.

Tel 0865 736488
Fax 0865 736835

Mill Street News

New Catalogue

As revealed in the last Newsletter, RM have indeed released their new Schools catalogue. This timely tome contains a wealth of information about RM products, including the latest prices and ordering methods.

Therein lies the bad news. Despite their best efforts to avoid price increases, RM have had to respond to the huge increase in memory chip prices world-wide and prices for all micros have increased. Details have been sent to all schools, but basically, .5Mbyte machines go up by £35 and 1Mbyte by £70 - at the moment. As previously announced, RM are not selling a 1.5Mbyte configuration for the time being.

Elsewhere in this issue is an add from Whittle systems which may help alleviate the problem - if you have the money. I understand that chips that cost £2.50 a year ago are, in some cases, being sold at £12. There is no indication that prices will drop this year.

The good news lies on the software and sales front for RM and its customers. Sales are booming, sort of, and the evidence is that RM have overtaken Acorn as the leading supplier to education.

Sales

Sales for the period Oct 87 to Mar 88 were £21 million. This is double last year's performance and well above target.

Narrowboat

On your holidays in the UK, look out for the ***Made in Britain*** 70-foot narrowboat. This was launched in April on a 7-month cruise of inland waterways promoting British-made products in 29 towns and cities. On board is - you've guessed it - (no, not a 380Z) an AX system to maintain the database. This has been lent by RM. Rumours that the boat is captained by Colin Watkins have been denied.

Computing Quiz

RM sponsored the national final of the BCS/Computer Education Group Schools Computing Quiz for 1988. The top prize is an AX with a PC-186 as the second prize. The final took place on July 2nd at UMIST.

Time, gentlemen, please!

Last orders for the 480Z are required by the end of September. From 1st October, RM will not accept orders for complete systems. Upgrades, options, accesories, software, etc., will continue to be available from RM subject to availability. Spare parts and expertise will remain on tap and the User Group will always be here!

Maintenance

Micro Systems Maintenance (MSM) have moved their Rochdale depot to Bury and opened a new one in Birmingham. Details from MSM on 0793-616888

 © RM User Group 1988

Scotland goes RM

According to RM, the Nimbus is making great strides in Scotland with Tayside, Grampian and Borders buying lorryloads of networks. Other LEAs there are investigating them.

New AutoCAD Catalogue

Autodesk Ltd have announced the *AutoCAD Applications and Services Catalogue*. This is available for £3.50 (incl). Contact Autodesk on 01-928-7868

Price Reductions!

Yes folks, not everything goes up.

RM have reduced prices of external hard disks for the PC-186.

19" high-res monochrome monitors have been discontinued by RM and are being sold off at reduced prices.

The AX/VX External Cartridge Tape Drive is reduced by £260.

Floppy disks for AX/VX are reduced to incredibly high prices.

Printer News

Brother printers seem to be a supply problem and the 1109 is currently replaced by the 1209 - except that by the time you read this position should be reversed!

The Qume ScripTen Postscript laser printer in reduced by £1000 to £3495 - from RM. It can be obtained more cheaply elsewhere. Try Northamber on 01-391-2066.

An excellent alternative which is cheaper and faster is the QMS PS-810.

VX-386 Windows 386

RM are now back-shipping Microsoft Windows/386 which was(n't) supplied to all purchasers of VX systems from the 1st November 1987. Upgrades are available for customers who purchased Vxs before this date.

PC-186 Business Data Processing Pack

Designed to support GCSE criteria and the whole concept of teaching computing through case studies, this important pack provides a wealth of practical material. It costs £74.95 and comes from:

Resource, Exeter Road, Coventry Grove, Wheatley, Doncaster. DN2 4PY Tel: 0302-63800/63784.

PC-186 Business Game

Also from Resource, (above), is a versatile Business Game generator which costs a mere £12.95.

PC/AX/VX Oriel Database for Windows

Known in the trade as *Son of Quest*, Oriel is the latest (and, some say, the greatest) from the AUCBE stable. What I like is the fact that it uses Windows and is cheap. It costs £95 for commercial users, £65 for an educational site licence and just £40 for a single home or school user.

CLIPPER - A dBASE COMPILER

by Richard A. Naylor

Ashton-Tate's dBASE III Plus seems to be regarded as the standard microcomputer database management system against which to judge all others. In spite of many good competitors it still holds a powerful position in the market place. However, the dBASE language, even now, has a number of annoying deficiencies which programmers in more conventional languages would readily miss. The consequences of this has been the spawning of a variety of dBASE clones which seek to address some of these problems as well as ride on the back of the successful dBASE language.

When I first started using dBASE III Plus some six months ago I found the lack of proper array handling a particularly glaring fault. dBASE has a facility called macro substitution which enables arrays to be simulated but this is both clumsy, slow in processing and limited if you require large arrays. Alternatively you can set a database to simulate arrays but this can make the processing rather disk bound.

A second dislike that I identified almost immediately was to do with validation of input data. dBASE has good facilities for setting up screen forms for data input where the user can flick around the input fields entering and editing data.

This is achieved by the @..SAY....GET clause. Some data validation, such as length, type and numeric range checking is achieved by adding PICTURE and RANGE commands. However if you want other forms of validation such as a lookup check then it has to be done after screen entry is complete. This means that screens have to be redisplayed if data is invalid.

To share files on a Network in a multi-user application dBASE requires an expensive upgrade.

dBASE in the form supplied by Ashton-Tate is an interpreted language which for some purposes can be quite slow.

On the market are at least three products which seek to address problems in the dBASE system as defined by Ashton-Tate. These are the dBASE compilers *QUICKSILVER*, *FOXBASE* and *CLIPPER*. All these products have also introduced their own extra features making their versions of the language non-standard. However this seems a price many are prepared to pay for greatly improved performance.

I decided that a compiler was required when I was landed with a school admin problem ("More fool you!" - I hear you say).

 © *RM User Group 1988*

One task required was the production of a clash table. I remember writing such a program to do this in Turbo Pascal for the 480z which processed about 200 pupil records into a 40x40 table in the time taken to read the data off the disc - about 30 seconds.

I was dumbfounded when dBASE, using the same rather inefficient algorithm took $2^1/_2$ hours to process just 15 records. The *Clipper* compiler using arrays instead of macro substitution and a database table reduced the time to less than 2 minutes making an acceptable 30 minutes for some 230 records. A further substantial improvement was achieved by improving the efficiency of the algorithm such that the job is now done in less than 4 minutes and that includes printing out the table!

CLIPPER Is a true compiler generating object code which is then linked to the supplied libraries. Compilation appears to be fast with linking relatively slow. The supplied linker is PLINK86 which permits the linking of overlay programs but if overlays are not required then the standard Microsoft linker LINK is much quicker. The resulting EXE files are big (160k plus). A simple menu driven application demonstrating record addition, deletion, editing and enquiries would probably compile to about 180k.

Both the compiler and linker will work happily in the normal Nimbus environment but the EXE files produced require the IBM emulator to be running. So far I have only used the emulator in monochrome mode as I have not had the time to sort out the colours for the best effect. Code is provided which when linked in allows the programs generated to run on an ANSI based computer without the emulator so it would seem possible to construct a screen driver for the Nimbus.

Some of the most notable features of Clipper are:

1) As well as DO..WHILE..ENDDO, CASE..ENDCASE and multi line IF..ELSE..ENDIF Clipper adds FOR..NEXT to the control constructs.

2) User defined functions.

3) The VALID command used in on screen data entry with @..SAY..GET. When combined with user defined functions data validation is simplified and greatly improved.

4) Single dimension array handling with useful functions for filling, sorting and searching arrays. Two dimensional arrays are easily implemented.

5) Light bar menu selection and a special feature which permits easy setting up of pulldown menus and scrollable menus.

6) Simple to use file and record locking functions for multiuser applications on a network provided at no extra cost and not requiring any extra software.

7) Your compiled programs can be distributed without the need for a special licence.

8) Programs execute fast! (See below).

9) A program debugger can be linked in and amongst the utilities is an important one for creating files and indexes. Another utility is provided for report form and label generation. The utilities are provided in source code (Clipper) form which you compile yourself.

As programming languages go Clipper is not too bad. Except for the added features it is very compatible with dBASE III Plus and it provides most of the features that facilitate good structured programming.

Perhaps the one advantage of dBASE II Plus is the easier program development environment of an interpreter. The availability of the "dot" prompt where dBASE III can be used in command mode is very useful for trying out commands that are to be included in the programs.

Clipper does come with a program called DOT which gives access to a limited range of commands at the familiar "dot" prompt. . The manual is a good detailed reference guide with useful examples for each command and function. If you have not used dBASE before then one of the many books available would be essential in order to see how the language is best used.

Clipper can be obtained for about £395 but I gather that the producer, *Nantucket*, is shortly going to announce a significant deal to education.

Some speed comparisons between dBASE III Plus and Clipper:-

a)

```
* Array Demo Program
* Implement an array using macro substitution
* 200 elements initialised 5 times
? TIME()
x=0
DO WHILE x<5
 x=x+1
 i=0
 DO WHILE i<= 200
  i=i+1
  p=LTRIM(STR(i,3))
  num&p=i
 ENDDO
ENDDO
? TIME()
```

 © RM User Group 1988

dBASE III (interpreted) = 100 secs
Clipper (compiled) = 8 secs

b)

```
* Array Demo
* Using extra features of Clipper
? TIME()
DECLARE array[200]
FOR x=1 TO 5
 FOR i=1 to 200
  array[i]=i
 NEXT
NEXT
? TIME()
```

Clipper (compiled) = 4 secs

Methodist College Belfast

New

Primary and Lower Secondary Projects for the RM Nimbus

Archipelago

Quest for treasures on the islands of the Lost Sea
£17.50

Voyage to the Stars

Informative and imaginative journeys through the stars of the Milky Way
£17.50

Details from:

Aurora Educational Software,
5, Hawkwood Mount,
London E5 9EQ

Tel: 01-806-1752

 © RM User Group 1988

USING dBASE III+/II IN COMPUTER STUDIES

by Richard A. Naylor,

For many Computer Studies teachers the reduction in emphasis on traditional BASIC programming that the advent of GCSE has encouraged is undoubtedly welcome. The change in approach has permitted a much more varied, relevant and meaningful use of the computer in both practical and coursework with the needs and interests of all abilities being more appropriately met.

However, with the first examinations approaching and as many of us struggle to come up for air (for the third time?) it is clear that there has been a price to pay. Not only do we have to retain our familiarity with programming but we also have to be skilled practictioners in the use of spreadsheets, wordprocessors, database systems, viewdata and other packages.

For many this has been an almost overwhelming task. This is particularly so when it appears, at least in my experience, that at the start of GCSE very few teachers had any experience beyond programming.

Following this rapid change of emphasis I suspect that some exam boards may wish to review the balance between practical work and syllabus content in the light of the fact that the process of familiarising pupils with these packages to a depth sufficient for meaningful coursework to take place is itself a very time consuming business.

I describe here one package which I have found particularly fruitful.

dBASE III+

I have used the dBASE III+ II and dBASE III+ III database management systems to good effect during the past year or so. To my mind it is a system which is suitable for solving problems at various levels of sophistication.

Firstly it can be used like QUEST in command mode. Datafiles can be created and all the usual file handling operations such as adding, deleting and amending of records can be performed at the command prompt. Databases can be interogated, listed, sorted and indexed in a wide variety of ways. For added sophistication databases may be related on common fields in a simple manner. dBASE III+ III provides a pulldown menu system as a front end to allow the user to execute commands without the need to know their syntax in detail. When used in command mode it makes a particularly effective way of introducing pupils to all the common file processing operations. Batch processing can be very easily simulated.

 © RM User Group 1988

Secondly the commands may be combined into command files or programs. Program control constructs such as DO..WHILE, IF.. ELSE..ENDIF and CASE are provided. In dBASE III+ III procedures with parameter passing are available. All in all quite a respectable programming language which allows stand-alone applications to be produced similar to many of the old GCSE projects. One of my more able students, with no programming experience prior to GCSE, has produced a respectable video library loan system using two related database files and which does not require the user to have any knowledge of dBASE III+.

dBASE III+ is a package that can meet the needs of problem solving for a range of pupil abilities. dBASE III+ is regarded as a fourth generation language (whatever that is) so those who still wish to program can do so. Indeed, from what I can gather, programming is still the major approach to implementing solutions to real problems out there in the big wide world. I was surprised to learn recently that a local computer consultancy with some significant contracts was still using BASIC for many of its applications. Programming is not dead yet!

Below are ten worksheets which serve as an introduction to dBASE III+/II. They could be readily extended and adapted for dBASE III+/II. I have also included a simple standalone database system written in dBASE III+ which illustrates some of the basic file processing concepts.

USEFUL dBASE III+ BOOKS

I have found Computer Bookshops of Birmingham an extremely efficient supplier of a wide range of computer books. They give a discount to education and books often arrive the next day after a telephone order.

Working with dBASE III+ II by M. DePace,
pub Collins ISBN 0-00-383251-1

Working with dBASE III+ III by M. DePace
pub Collins

Using dBASE III+ III Plus by E. Jones
pub McGraw Hill ISBN 0-07-881252-6

Well, the modules to which Richard refers would run for some 20 pages of the Journal and I have chopped them off! However, they will go into the Software Library as an MSDOS volume. This disk should be available from Sheila Maynard soon after you read this and if you're using - or considering using - dBase II or III+ then they will be a great value. I have at least left you the sample database to puzzle or drool over. **Editor**

MUSIC DATABASE - A SAMPLE APPLICATION

1) Database Structure

```
Structure for database: C:MUSIC.dbf
Number of data records:     6
Date of last update  : 15/01/88
Field  Field Name  Type       Width    Dec
    1  NUMBER     Numeric        4
    2  TITLE      Character     30
    3  ARTIST     Character     30
    4  CLASS      Character     12
    5  MEDIA      Character      6
    6  CAT        Character     12
    7  COST       Numeric        5      2
** Total **                    100
```

2) Main Menu

```
* Music Collection Database
* Program: MUSIC.CMD
* File   : MUSIC.DBF
* Started      : 21/12/87
* Last Modified : 21/12/87
*************************************************************
* Set up program environment
*
SET TALK OFF
SET STATUS OFF
SET SAFETY OFF
*************************************************************
* Main Program
*
* Do menu forever!
DO WHILE .T.
 * Display menu screen
 CLEAR
 STORE " " TO mchoice
 @ 4, 23 SAY "MUSIC COLLECTION DATABASE"
 @ 7, 24 SAY "0) Finish with database"
 @ 9, 24 SAY "1) Add a new record"
 @ 10, 24 SAY "2) Delete a record"
 @ 11, 24 SAY "3) Edit a record"
 @ 12, 24 SAY "4) Title enquiry"
```

 © RM User Group 1988

```
 @ 13, 24  SAY "5) Artist/Group enquiry"
 @ 14, 24  SAY "6) List Database"
 @  2, 15  TO 19, 55   DOUBLE
 * Do until a valid choice is entered
 DO WHILE .NOT. (mchoice $ "0123456")
  STORE " " TO mchoice
  @17,24 SAY "Enter choice : " GET mchoice PICTURE "!"
  READ
 ENDDO
 *
 * Execute the choice
 DO CASE
  CASE mchoice = "0"
   CLEAR
   SET STATUS ON
   SET TALK ON
   SET SAFETY ON
   RETURN
  CASE mchoice = "1"
   DO add
  CASE mchoice = "2"
   DO del
  CASE mchoice = "3"
   DO musedit
  CASE mchoice = "4"
   DO enqtitle
  CASE mchoice = "5"
   DO enqart
  CASE mchoice = "6"
   DO muslist
 ENDCASE
ENDDO
************************************************************
```

3) Add a Record

```
* Music Collection Database
* Program: ADD.CMD
* File   : MUSIC.DBF
* Add a record program
* Started      : 20/12/87
* Last modified : 21/12/87
*********************************************************************
* Set up program environment
*
```

 © RM User Group 1988

```
SET TALK OFF
SET HEADING OFF
SET SAFETY OFF
*********************************************************************
* Set up database for use with all its indexes
* This is so that the indexes are automatically updated when a
* new
* record is added.
* Index number is the main index.
*
USE music INDEX number,title, artist
*
*********************************************************************
* Main Program
*
CLEAR
STORE .T. TO again
*
* Repeat until user has no more records to add
DO WHILE again
 * Routine to allocate a unique number to each record added  STORE 0 TO mnumber
 GOTO BOTTOM
 * Add one to get number for the new record about to be
 * added
 STORE number+1 TO mnumber
 * Set up memory variables
 STORE "                    " TO mtitle
 STORE "                    " TO martist
 STORE "          " TO mclass
 STORE "    " TO mmedia
 STORE "          " TO mcat
 STORE 0 TO mcost
 *
 STORE .F. TO correct
 *
 * Repeat data entry until user says it is correct
 DO WHILE .NOT. correct
  *
  * Data entry screen
  *
  @ 2, 23 SAY "MUSIC DATABASE - ADD A NEW RECORD"
  @ 4, 15 SAY "Number    :"
  @ 4, 28 say mnumber
  @ 6, 15 SAY "Title     :"
  @ 6, 28 GET mtitle;
```

 © RM User Group 1988

```
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
 @  8, 15  SAY "Artist    :"
 @  8, 28  GET  martist;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
 @ 10, 15  SAY "Class     :"
 @ 10, 28  GET  mclass PICTURE "!!!!!!!!!!!!"
 @ 10, 42  SAY "(POP, CLASSICAL, RELIGIOUS, FOLK)"
 @ 12, 15  SAY "Media     :"
 @ 12, 28  GET  mmedia PICTURE "!!!!!!"
 @ 12, 42  SAY "(CD, TAPE, LP, SINGLE)"
 @ 14, 15  SAY "Catalogue :"
 @ 14, 28  GET  mcat  PICTURE "!!!!!!!!!!!!!"
 @ 16, 15  SAY "Cost      :"
 @ 16, 28  GET  mcost  PICTURE "99.99"
 @  1, 18  TO  3, 60   DOUBLE
 @  0,  5  TO 20, 75
 READ
 *
 * If no title has been entered then finish program.
 IF mtitle = " "
   CLEAR
   RETURN
 ENDIF
 *
 * Get user to confirm data is correct as entered
 STORE " " TO answer
 DO WHILE .NOT. (answer $ "YyNn")
  STORE " " TO answer
  @ 18, 6 SAY "                                "
  @ 18, 6 SAY ;
  "          Is this correct? (Y/N) : " GET answer
  READ
  @ 18, 6 SAY "                                "
 ENDDO
 IF answer $ "Yy"
  STORE .T. TO correct
 ENDIF
ENDDO
*
* Add new record to database
*
APPEND BLANK
REPLACE number WITH mnumber
REPLACE title WITH mtitle
REPLACE artist WITH martist
```

```
REPLACE class WITH mclass
REPLACE media WITH mmedia
REPLACE cat WITH mcat
REPLACE cost WITH mcost
*
* Add another record ?
*
STORE " " TO answer
DO WHILE .NOT. (answer $ "YyNn")
  STORE " " TO answer
  @ 18, 6 SAY "                                    "
  @ 18, 6 SAY "          Add another record? (Y/N) : ";
        GET answer
  READ
  @ 18, 6 SAY "                                    "
ENDDO
IF answer $ "Nn"
  CLEAR
  STORE .F. TO again
ENDIF
*
ENDDO
************************************************************
```

4) Edit a Record

```
* Music Collection Database
* Program: MUSEDIT.CMD
* File   : MUSIC.DBF
* Edit a record program
* Started      : 21/12/87
* Last modified : 21/12/87
*****************************************************************
* Set up program environment
*SET TALK OFF
SET HEADING OFF
SET SAFETY OFF
*****************************************************************
* Set up database for use with all its indexes
* This is so that the indexes are automatically updated when a
* record is edited.
* Index number is the main index.
*
USE music INDEX number,title, artist
*
```

```
*********************************************************************
* Main Program
*
CLEAR
STORE .T. TO again
*
* Repeat until user has no more records to add
DO WHILE again
 CLEAR
 STORE 0 TO mnumber
 @ 2,30 SAY "EDIT A RECORD"
 @ 5,10 SAY "Enter Reference number of entry to be edited"
 @ 6,10 SAY "or press <RETURN> to finish : ";
        GET mnumber PICTURE "9999"
 @ 1,18 TO 3,60 DOUBLE
 @ 0,5 TO 20,75
 READ
 IF mnumber = 0
  CLEAR
  RETURN
 ELSE
  STORE STR(mnumber) TO mnumber
  FIND &mnumber
  IF EOF()
   @ 11,10 SAY;
     "Record Reference " + mnumber + " not found..."
  ELSE
   STORE number TO mnumber
   STORE title TO mtitle
   STORE artist TO martist
   STORE class TO mclass
   STORE media TO mmedia
   STORE cat TO mcat
   STORE cost TO mcost
   *
   STORE .F. TO correct
   *
   * Repeat data entry until user says it is correct
   DO WHILE .NOT. correct
   *
   * Data entry screen
   *
   CLEAR
   @ 2, 23 SAY "MUSIC DATABASE - EDIT A RECORD"
   @ 4, 15 SAY "Number    :"
```

 © RM User Group 1988

```
@ 4, 28 SAY mnumber
@ 6, 15 SAY "Title    :"
@ 6, 28 GET mtitle;
   PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@ 8, 15 SAY "Artist   :"
@ 8, 28 GET martist;
   PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@ 10, 15 SAY "Class    :"
@ 10, 28 GET mclass PICTURE "!!!!!!!!!!!!"
@ 10, 42 SAY "(POP, CLASSICAL, RELIGIOUS, FOLK)"
@ 12, 15 SAY "Media    :"
@ 12, 28 GET mmedia PICTURE "!!!!!!"
@ 12, 42 SAY "(CD, TAPE, LP, SINGLE)"
@ 14, 15 SAY "Catalogue :"
@ 14, 28 GET mcat PICTURE "!!!!!!!!!!!!!!"
@ 16, 15 SAY "Cost     :"
@ 16, 28 GET mcost PICTURE "99.99"
@ 1, 18 TO 3, 60  DOUBLE
@ 0, 5 TO 20, 75
READ
*
* Get user to confirm data is correct as entered
STORE " " TO answer
DO WHILE .NOT. (answer $ "YyNn")
 STORE " " TO answer
 @ 18, 6 SAY "                                "
 @ 18, 6 SAY;
 "          Is this correct? (Y/N) : " GET answer
 READ
 @ 18, 6 SAY "                                "
ENDDO
IF answer $ "Yy"
 STORE .T. TO correct
ENDIF
ENDDO
*
* Add new record to database
*
REPLACE number WITH mnumber
REPLACE title WITH mtitle
REPLACE artist WITH martist
REPLACE class WITH mclass
REPLACE media WITH mmedia
REPLACE cat WITH mcat
REPLACE cost WITH mcost
```

 © RM User Group 1988

```
  *
 ENDIF
 * Edit another record ?
 *
 STORE " " TO answer
 DO WHILE .NOT. (answer $ "YyNn")
  STORE " " TO answer
  @ 18, 6 SAY "                                    "
  @ 18, 6 SAY;
   "          Edit another record? (Y/N) : " GET answer
  READ
  @ 18, 6 SAY "                                    "
 ENDDO
 IF answer $ "Nn"
  CLEAR
  STORE .F. TO again
 ENDIF
 *
ENDDO
*********************************************************************
```

5) Delete a Record

```
* Music Collection Database
* Program: DEL.CMD
* File   : MUSIC.DBF
* Delete a record
* Started      : 21/12/87
* Last Modified :
*********************************************************************
*Set up Program environment
*
SET TALK OFF
SET HEADING OFF
SET SAFETY OFF
*********************************************************************
* Set up database for use
*
USE music INDEX number,title,artist
*********************************************************************
* Main Program
DO WHILE .T.
 CLEAR
 STORE 0000 TO mnumber
 @ 2,30 SAY "DELETE A RECORD"
```

```
@ 5,10 SAY "Enter Reference number of entry to be deleted"
@ 6,10 SAY "or press <RETURN> to finish : ";
    GET mnumber PICTURE "9999"
@ 1,18 TO 3,60 DOUBLE
@ 0,5  TO 20,75
READ
IF mnumber = 0
 CLEAR
 RETURN
ELSE
 STORE STR(mnumber) TO mnumber
 FIND &mnumber
 IF EOF()
  CLEAR
  @ 11,10 SAY "Record Reference "+ mnumber +" not found..."
 ELSE
  * Display record to be deleted
  *
  CLEAR
  @  2, 23  SAY "MUSIC DATABASE - DELETE A RECORD"
  @  4, 15  SAY "Number    :"+STR(number)
  @  6, 15  SAY "Title     :"+ title
  @  8, 15  SAY "Artist    :"+ artist
  @ 10, 15  SAY "Class     :"+ class
  @ 12, 15  SAY "Media     :"+ media
  @ 14, 15  SAY "Catalogue :"+ cat
  @ 16, 15  SAY "Cost      :"+ STR(cost)
  @  1, 18  TO  3, 60   DOUBLE
  @  0,  5  TO 20, 75
  *
  * Get user to confirm this is record to be deleted
  STORE " " TO answer
  DO WHILE .NOT. (answer $ "YyNn")
   STORE " " TO answer
   @ 18, 6 SAY "                                "
   @ 18, 6 SAY;
   "            Delete this record? (Y/N) : " GET answer
   READ
   @ 18, 6 SAY "                                "
  ENDDO
  IF answer $ "Yy"
   STORE .T. TO correct
   DELETE
   PACK
   CLEAR
```

 © RM User Group 1988

```
    @ 11,10 SAY "Record Reference "+mnumber+" DELETED..."
   ENDIF
  STORE " " to manswer
  @ 19,10 SAY "Press <RETURN> to continue.." GET manswer
  @ 0,5 TO 20,75
  READ
 ENDIF
ENDDO
```

6) Title Enquiry

```
* Music Collection Database
* Program: ENQTITLE.CMD
* File   : MUSIC.DBF
* Title enquiry program
* Started      : 12/12/87
* Last Modified : 21/12/87
**********************************************************************
* Keywords in upper case, identifiers in lower case
* Memory variables begin with 'm' usually
**********************************************************************
* Set up program environment
*
SET TALK OFF
SET EXACT OFF
SET HEADING OFF
SET SAFETY OFF
**********************************************************************
* Set up database for use with index
*
USE music INDEX title
**********************************************************************
* Main part of program
*
CLEAR
DO WHILE .T.
 * Display screen form requesting title to search for
 @ 2,30 SAY "TITLE ENQUIRY"
 STORE "                    " TO mtitle
 @ 5,10 SAY "Enter Title :" get mtitle;
        PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
 * The above picture clause forces entry into upper case
 @ 6,10 SAY "or <RETURN> to finish"
 * Draw boxes on screen
 @ 1,18 TO 3,60 DOUBLE
```

```
@ 0,5 TO 20,75
* Read the data typed at the keyboard
READ
* Exit program if only <RETURN> typed
IF mtitle = " "
 CLEAR
 RETURN
ELSE
 * Else search index for the title required
 FIND &mtitle
 IF EOF()
  * If end of file reached then title does not exist
  @ 11,10 SAY "Title "+mtitle+"not found.."
 ELSE
  * Else display the details of the title found
  CLEAR
  @ 0,5 TO 20,75
  @ 3,10  SAY "Ref Number  : " + STR(number,4)
  @ 5,10  SAY "Title       : " + title
  @ 7,10  SAY "Group/Artist : " + artist
  @ 9,10  SAY "Type        : " + class
  @ 11,10 SAY "Media       : " + media
  @ 13,10 SAY "Catalogue No : " + cat
  @ 15,10 SAY "Price paid  : " + STR(cost,5,2)
 ENDIF
 * Press return routine to allow user to read screen
 STORE " " TO manswer
 @ 19,10 SAY "Press <RETURN> to continue " get manswer
 READ
 CLEAR
ENDIF
ENDDO
*****************************************************************
```

7) Artist Enquiry

```
* Music Collection Database
* Program: ENQART.CMD
* File   : MUSIC.DBF
* Title enquiry program
* Started      : 15/12/87
* Last Modified : 21/12/87
*****************************************************************
* Keywords in uppercase, identifiers in lowercase
* All memory variables begin with an 'm'
```

 © RM User Group 1988

```
*********************************************************************
* Set up the program environment
*
SET TALK OFF
SET EXACT OFF
SET HEADING OFF
SET SAFETY OFF
*********************************************************************
* Set up database for use with index
*
USE music INDEX artist
*********************************************************************
* Main Program
*
CLEAR
DO WHILE .T.
 *
 * Display request form for artist to be searched for.
 *
 @ 2,30 SAY "ARTIST/GROUP ENQUIRY"
 STORE "                              " to martist
 @ 5,10 SAY "Enter Artist/Group :" GET martist;
       PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
 @ 6,10 SAY "or <RETURN> to finish"
 *
 * Draw boxes on screen
 *
 @ 1,18 TO 3,60 DOUBLE
 @ 0,5 TO 20,75
 *
 * Read data from keyboard according to screen display
 *
 READ
 * Exit program if only <RETURN> typed
 IF martist = " "
  CLEAR
  RETURN
 ELSE
  * Else search for the artist typed in
  FIND &martist
  * If end of file found then the artist does not exist
  IF EOF()
   CLEAR
   @ 11,10 SAY "Artist/Group "+martist+" not found.."
   * Press return to continue routine so that screen may be
```

```
    * read
    STORE " " TO manswer
    @ 19,10 SAY "Press <RETURN> to continue.." GET manswer
    @ 0,5 TO 20,75
    READ
   ELSE
    * Artist found so display all records that match for this
    * artist
    * until we get to end of file
    DO  WHILE martist $ artist .AND. .NOT. EOF()
     CLEAR
     @ 0,5 TO 20,75
     @ 3,10  SAY "Ref Number   : " + STR(number,4)
     * Numeric field must be converted to string
     @ 5,10  SAY "Title        : " + title
     @ 7,10 SAY  "Group/Artist : " + artist
     @ 9,10  SAY "Type         : " + class
     @ 11,10 SAY "Media        : " + media
     @ 13,10 SAY "Catalogue No : " + cat
     @ 15,10 SAY "Price paid   : " + STR(cost,5,2)
     * Move to next record
     SKIP
     * Press return to continue routine
     STORE " " TO manswer
     @ 19,10 SAY "Press <RETURN> to continue.." GET manswer
     READ
     CLEAR
    ENDDO
   ENDIF
   CLEAR
  ENDIF
ENDDO
*********************************************************************
```

8) Database Listings

```
* Music Collection Database
* Program: MUSLIST.CMD
* Produces Listings to screen or printer of the database
* File   : MUSIC.DBF
* Started      : 22/12/87
* Last Modified : 22/12/87
*********************************************************************
* Set up program environment
*
```

 © RM User Group 1988

Whittle Systems

For
YOUR NIMBUS PC
Expansion

APPROVED
CUMANA DISTRIBUTOR

MEMORY

Is your Nimbus PC186 running out of memory?

Do you need **1 Mega Byte** plus to run the latest software?

Do you want to be able to run effectively the NET2 Menu System, Windows, the IBM Emulator, PaintSPA, NewSPAper, DataEase... ?

Fed up with waiting?

Memory boards available from stock!

R0	0 k byte blank memory board, find your own chips	**£45.00**
R5	512 k byte memory board, (2nd half socketed, so you can upgrade later)	**£175.00**
R10	1024 k byte memory board.	**£325.00**

Hurry whilst stocks last! *First come first served.*

All prices exclude P & P and Vat.

53, Besselsleigh Road,
Wootton,
Abingdon, Oxford OX13 6DX.

Tel 0865 736488
Fax 0865 736835

Whittle Systems

For
YOUR NIMBUS PC
Expansion

APPROVED
CUMANA DISTRIBUTOR

MEMORY

Is your Nimbus PC186 running out of memory?

Do you need **1 Mega Byte** plus to run the latest software?

Do you want to be able to run effectively the NET2 Menu System, Windows, the IBM Emulator, PaintSPA, NewSPAper, DataEase... ?

Fed up with waiting?

Memory boards available from stock!

R0	0 k byte blank memory board, find your own chips	**£45.00**
R5	512 k byte memory board, (2nd half socketed, so you can upgrade later)	**£175.00**
R10	1024 k byte memory board.	**£325.00**

Hurry whilst stocks last! *First come first served.*

All prices exclude P & P and Vat.

53, Besselsleigh Road,
Wootton,
Abingdon, Oxford OX13 6DX.

Tel 0865 736488
Fax 0865 736835

```
SET TALK OFF
SET HEADING OFF
SET SAFETY OFF
*********************************************************************
* Main Program
*
* Do menu forever!
STORE " " TO out
DO WHILE .T.
 * Display menu screen
 CLEAR
 STORE " " TO mchoice
 @  4, 23  SAY "MUSIC COLLECTION DATABASE"
 @  5, 23  SAY "   DATABASE LISTINGS"
 @  7, 21  SAY "0) Return to Main menu"
 @  9, 21  SAY "1) Send Listing to "
 @ 10, 21  SAY "2) Alphabetic List by Title"
 @ 11, 21  SAY "3) Alphabetic List by Performer"
 @ 12, 21  SAY "4) List in Reference Number Order"
 @ 13, 21  SAY "5) Full Listing of Database"
 IF out=" "
  @ 9,40 SAY "PRINTER"
  @ 15,24 SAY "Output is to SCREEN "
 ELSE
  @ 9,40 SAY "SCREEN"
  @ 15,24 SAY "Output is to PRINTER"
 ENDIF
 @  2, 15  TO 19, 55    DOUBLE
 * Do until a valid choice is entered
 DO WHILE .NOT. (mchoice $ "012345")
  STORE " " TO mchoice
  @17,24 SAY "Enter choice : " GET mchoice PICTURE "!"
  READ
 ENDDO
 *
 * Execute the choice
 CLEAR
 DO CASE
  CASE mchoice = "0"
   RETURN
  CASE mchoice = "1"
   IF out=" "
    STORE "TO PRINT" TO out
   ELSE
    STORE " " TO out
```

 © RM User Group 1988

```
    ENDIF
  CASE mchoice = "2"
    USE music INDEX title
    REPORT FORM music1 HEADING "Alphabetic List by Title" &out
    CLOSE ALL
  CASE mchoice = "3"
    USE music INDEX artist
    REPORT FORM music1;
        HEADING "Alphabetic List by Performer" &out
    CLOSE ALL
  CASE mchoice = "4"
    USE music INDEX number
    REPORT FORM music1;
        HEADING "List by Reference Number" &out
    CLOSE ALL
  CASE mchoice = "5"
    USE music INDEX artist
    REPORT FORM music2 &out
    CLOSE ALL
 ENDCASE
ENDDO
*****************************************************************
```

Methodist College Belfast

 © RM User Group 1988

Authoring shells - the next generation of content free software?

I wouldn't like to have to count the hours I have spent producing teaching materials of various sorts over the last 17 years. Most of the materials have been paper based, and have since been superseded by new bits of paper (and more hours work!). When micros first appeared in school I had the idea that I might produce some software, which would serve as 'interactive' courseware, and could be updated with less effort than rewriting paper based materials.

Although I learned BASIC I soon discovered that I had neither the expertise nor the time to produce the software/coursware that I wanted. What I really needed was some sort of authoring shell which would allow me to 'program' at a high level. Such a shell would only require me to supply the data, and it would do the rest, making an interactive piece of software. At that time I don't think such things existed, and if they did they certainly wouldn't run on a Sinclair Spectrum or a 380Z!!

Such pieces of software do exist now, and I have been using them over the last 18 months or so. These software shells or authoring shells (also now being called Knowledge Based Systems) all happen to be written in Micro-Prolog, and run on the Nimbus. This doesn't mean that you need to know anything about Micro-Prolog to use the shells, a working knowledge of English is all that is required! The two pieces of software I have been using are :-

LINX++ - A simulations writer with attached database
DETECT - A detective database writer

The shells allow the teacher to program simulations and detective (and other) databases with no programming knowledge, and hence make them wholly relevant to the work being covered. I have used them in conjunction with the headmaster and his class of 7 and 8 year olds at Randwick Primary School, Stroud in Gloucestershire. I wanted to investigate the potential of the software for supporting the normal curriculum of the class. The approach I took was to ask the teacher what the class would be covering in the near future and then discuss how we could use the micro to enhance the work. We would then formulate a detailed simulation (or whatever) and type it in to the shell.

We used the software alongside the class project work which followed the BBC schools programmes Zig Zag, and when we started this work they were looking at life as an Eskimo! During this topic we used LINX++ to produce a mini expert system on how to build an igloo. The children worked on the program in small groups the object being to give them a decision making task in trying to build the igloo. They had to decide where would be the best place, what type of snow to use, how to start it off, where to put the window etc. The program worked well producing some quality discussion on such topics as what happens to hot air, and how do you prevent predators entering the igloo!

 © RM User Group 1988

We thought that it might be a good idea to see if the children could write their own 'programs'. I wrote a simple simulation using LINX++ (with only about 6 'locations') and let the children use it. I then showed them the structure of the 'program' and suggested that they have a go at writing a branching story which we could then type in to the computer.

The children found this very difficult - after all most of the stories that they had written so far only had one ending. If only because it made them write a different type of story, and consider other possible outcomes of events, this was worth doing - it made the children look at story writing in a completely different way. We continued the theme of trying to get the children to write a program, and by this time Zig Zag had moved onto the Canadian Pacific Railway. We spent a whole afternoon discussing the problems that might have faced the builders of the railway, and what choices were open to them, and what the outcomes of those choices were likely to be. At the end of the afternoon we had a sort of gigantic flowchart across the 3 blackboards. Apart from that- we had nothing else!

There is no denying we had had a very useful discussion and a great deal of thinking had taken place - but we didn't have a lot to show for an afternoon's work! I needn't have worried about it - when I returned to the school the following week the 'flowchart' had been used to produce a wall display with scenes painted by the children depicting each potential problem that might crop up in constructing the railway with the possible results, all the paintings being linked into a giant flowchart.

We have also used the DETECT shell to produce course material. The first use that we made of this was to write a murder mystery set in the North American goldrush. The idea was to give the children a flavour of what life might have been like at the time. We used authentic names for our characters taken from the Zig Zag materials. The Ping Pong Kid had been killed! Shot in the back outside the Lucky Strike saloon in Dawson City. The children could investigate the crime by using the following :

```
tell about [person or place]
interview  [person]
search     [place]
```

At first they found it hard going, being much more open ended than the work with LINX++. However, they soon got the hang of investigating the crime, and came to a unanimous conclusion - they needed more hard evidence. They suggested that being able to examine fingerprints might be a good idea, and such is the ease of use of the shell that I was able to add a facility for checking objects to see if they had any fingerprints on them during the lesson!

The software did have some failings, for example it was not as easy to edit as it might have been. However, problems which we discovered during our use in the classroom have been fed back to the programmers working for PEG Exeter and the latest software overcomes these problems. Since the Ping Pong Kid we have produced courseware for a topic on the

 © RM User Group 1988

Crusades, our school trip, a viking village dig, an adventure on the planet Gargantua, and a tour of the town of Dimville which we have used to develop map making skills with the children.

Next term we hope to produce some materials based on the Olympic games. The positive features of the software are that there is a common environment for both the shells so that if you can use one then transferring to the other is easy. Both the shells are flexible and can be used for a variety of purposes, and any editing is now extremely straightforward so making files extensible in the classroom - a very powerful feature - allowing the courseware to develop with the lesson if required.

As authoring tools such as these develop they will take their place alongside other content free forms of software such as wordprocessors, databases and spreadsheets.

A note on the terms used.

I've tried to reserve the word 'software' for the authoring programs themselves i.e. LINX++ and DETECT.

I've also used the term 'shell' to describe the software (LINX++ and DETECT) as they provide a working environment which allows the user to produce what appears to be a program.

The 'program' produced I've called 'coursware' as it will be relevant to a particular course or section of the curriculum.

I've talked about 'authoring' - meaning that the user of the shell is using it to author the coursware (which is in fact a file).

PEG Exeter - This used to mean Prolog Education Group Exeter as the shells which they have produced are written in PROLOG. It now means something else the P reperesenting PICKET (which is a word made up using the first letter of lots of things!) - so now PEG = Picket Education Group - doesn't life get complicated?

If you would like further information on the latest versions of this software now called LINX88 and KEYS88 please contact :

Dr. Jon Nicol,
PEG Exeter,
Dept. of Education,
St. Lukes,
University of Exeter,
Exeter.

or me,
Les Watson,
Dept. Maths Science and Computing,
College of St. Paul and St. Mary,
The Park,
Cheltenham,
Gloucestershire,
GL50 2RH

If you would like to use the software I shall be doing a presentation at the MUSE conference in Nottingham in July, at the BLUG conference in Cheltenham in September, and at the RM user group conference in October.

Writing for the Journal

Please do! All contributions are welcomed and most are published, subject to editorial requirements. Your contributions should be sent on disk from any RM machine and it may be helpful in some cases to include a printout of the article. All disks are eventually returned!

One of the greatest needs is for suitable illustrations, but these need to be disk files which can be transferred into *Aldus PageMaker*. *Windows Draw!* pictures are the best, but *PC Paintbrush* pictures are acceptable. If you want to send in graphics from such as PAINT*SPA* or NEW*SPA*PER, please print out with a new ribbon so that there is some chance of making camera-ready copy.

For those who would really like to help the Editor and are using a Nimbus and either *Microsoft Word* or *Windows Write!*, please format your text in 9 point Times Roman. When using *MS Word* you will need to select a postscript printer (Apple laser) from the Print Options menu. Do **not** use a style sheet (for various reasons!)

If you are using, say, *WordStar* on a 380Z or 480Z (*and why not*?) then do it as a non-document file. I cannot read BBC disks (although that should have changed by the time you read this), but you may be able to email me on 72:MAG 90380.

Don't be put off by any apparent technical depths in material you read in the Journal. We *eagerly* welcome simple, straightforward stuff from novices for novices. There is no minimum length, either! If you've an idea, but are uncertain as to its usefulness, please drop me a line or ring me. We are very keen on any material concerning BBCBASIC-86 and the running of BBC programs on the NIMBUS PC-186. Articles/letters about your experiences with RM, their kit and service are welcome, as are the same about other vendors of hardware or software for the RM range.

Deadline for the next Journal is September 9th 1988

 © RM User Group 1988

I T E M ilea
INFORMATION TECHNOLOGY EDUCATIONAL MATERIALS
Self-instructional guides -Perfect for Staff and Students
MS-WORD Complete with keyboard overlay
MULTIPLAN Suitable in HE/FE, Secondary and Primary,
LOGISTIX for CPVE, RSA, TVEI, YTS, GCSE, BTEC, CGLI
MS-DOS and Staff Development, GRIST, etc. More
DATAEASE than 40,000 Guides sold all over the UK.
Software for RM-Nimbus -Use IT across the Curriculum
MAILBOX Full feature electronic mail on a network
CHAT Station to station calls on a network
SIMTEX Delivers prepared telex messages
KEYSTROKE Keyboarding tutor in full colour
REVOLVER Local Viewdata with on-screen help
MULTIGRAPH Produces graphics from Multiplan and Quest files or direct keyboard input
New teaching packs -With files & worksheets available soon
Network versions and Licence agreements available as well as discounts on all materials for bulk purchases.
Ring, mbx or write for details and catalogue
Tel: 01-735-9123 or TTNS: TCD 100162
I T E M, ILECC, John Ruskin St., London, SE5 0PQ

 © RM User Group 1988

Nimbus - BASIC electronic communications

by Mark Bryson

In common with many other people, I find communications programs in general much less congenial than other types of application software such as word-processing or graphics; there are many reasons why this might be so, but I have yet to find a communications program that didn't (in my view) need considerable changes. However, to develop your own communications software, you've really got to use a language faster than BASIC and preferably one that gives greater access to the 'low-level' features of your micro - such programs are normally written in Assembler.

The programs listed below can not be considered an acceptable substitute for *proper* communications programs like KERMIT, CONTACT, DIALUP or a terminal emulator; although I hope they may help to de-mystify the workings of communications programs in general.

The programs run on Nimbus (186) micro with MSDOS 3.10 and RM BASIC 1.0F; although they could perhaps be translated for other dialects of BASIC or different languages on other MSDOS or CP/M micros. It seems that the presence of some memory resident programs may upset the working of these programs; in particular, if you see the BASIC error message, 'Wrong channel number used', it is worth switching off, re-booting, loading (just) BASIC and trying again.

I developed the programs using a Nightingale modem, which has an (local) echo facility intended to allow checking of communications software and hardware without incurring telephone charges. If your modem has a similar option it will certainly save you money!

The filename **COMB** (you can also use "AUX") referred to in the programs is the name of a serial connection found on all Nimbus PC186s (labelled AUX). Other micros may use other names for their serial port(s), usually depending on their operating system. Other MSDOS micros might use "COM1". Check your operating manuals for details.

The 480Z under CP/M uses "RDR:" for input and "LST:" for output. Years ago I used a similar technique to transfer files between a 480Z CHAIN network and an Econet.

It is vital to configure the AUX port to operate at a suitable speed for your modem and to use suitable flow control. Xon/Xoff, the CONFIG 'DX' option, is suitable. The Nimbus and Nightingale union requires a 300 baud setting (or about 30 characters/sec) but the Tandata TM512 requires 1200 baud.

The Tandata TM512 modem communicates with its micro at 1200 baud, but at this speed RM BASIC 1.0F is NOT fast enough (using AUTOMAIL.BAS) to catch all characters arriving at the AUX port from the modem. Presumably the AUX will 'shake hands' with

 © RM User Group 1988

the modem but not with BASIC? Since the AUX transmit and receive buffers each hold only 8 bytes many characters are 'lost'. NIMBCOMB.BAS can cope with the higher speed of transfer but lacks even the limited features of 'AUTOMAIL'.

It is possible to use programs like NIMBCOMB.BAS to control the Tandata modem to set communication speed (baud rate), make it dial (just type the number you want), connect and disconnect etc. The modem's responses to these commands can then be acted upon under program or manual control - just what 'proper' communications programs do.

The keypad keys are used to control program functions such as downloading and are redefined to produce specific values, for example the END key produces the ASCII value 255. The values used are not significant except they should obviously not be required for any other purpose, which excludes values in the range 32 to 126 and 13 because these represent ASCII text characters and <enter>.

There must be a better way to code these programs but if something works I tend to leave it alone; nevertheless, some features of RM BASIC 1.0F may require explanation:

- Variables suffixed by a $ character are string variables. In RM BASIC these may be many thousands of characters long. This is not the case with many other BASIC dialects, which might be a source of trouble in the 'Look_For' procedure, although such a problem can be easily solved.

- The PUT command is equivalent to PRINT without an automatic linefeed, for example: PUT 27 in RM BASIC is the same as PRINT CHR$(27); (with trailing semi-colon) in 'standard' BASIC.

```
 1 REM Nimcomb.bas 16.12.87
10 SET MODE 80
20 :
30 OPEN #20, "COMB"
40 C% := GET(#20, 1) : IF C% > 0 THEN PUT C% : GOTO 40
50 CLOSE #20 : K% := GET(#0, 1) : IF (K% AND 128) > 0 THEN
END
60 IF K% <> 0 THEN CREATE #20, "COMB" : PUT #20, K% : CLOSE
#20
70 GOTO 30
```

```
 1 REM AutoMail.bas 16.12.87
100 SET MODE 80
110 GLOBAL Ln_Fd, Pg_Dn, Pg_Up, Pr_Nt
120 Pg_Dn := FALSE : Pg_Up := FALSE : Pr_Nt := FALSE : Ln_Fd
  := TRUE
```

```
130 Re_Define_Keys
140 :
150 Send 13 : Send 10 : IF Chr_In() = 0 THEN 150
170 :
180 REM : Look_For "username:"
190 REM : Send_Str "fred"
200 REM : Look_For "password:" : Send_Str "secret"
210 :
220 REPEAT
230   Send Keyb() : Show Chr_In()
240   IF Pg_Dn THEN Download
250   IF Pg_Up THEN Upload
260 UNTIL FALSE
270 END
280 :
290 PROCEDURE Send C%
300 IF C% <> 0 THEN CREATE #20, "COMB" : PUT #20, C% :
  CLOSE #20
310 ENDPROC
320 :
330 FUNCTION Chr_In()
340 OPEN #20, "COMB" : C% := GET(#20, 1) : CLOSE #20 :
  RESULT C%
350 ENDFUN
360 :
370 FUNCTION Keyb()
380   GLOBAL Pg_Up, Pg_Dn, Pr_Nt, Ln_Fd
390   K% := GET(#0, 1)
400   IF K% = 255 THEN CLOSE : END
410   IF K% = 254 THEN Pg_Up = Pg_Up XOR TRUE : PRINT
    "*Upload*"
420   IF K% = 253 THEN Pg_Dn := Pg_Dn XOR TRUE : PRINT
    "*Download*"
430   IF K% = 252 THEN Pr_Nt := Pr_Nt XOR TRUE : PRINT
    "*PRINT*"
440   IF K% = 251 THEN Ln_Fd := Ln_Fd XOR TRUE : PRINT
    "*Line Feed*"
450   IF K% = 9 THEN K% := 32
460   IF K% > 127 THEN K% := 0
470   RESULT K%
480 ENDFUN
490 :
500 PROCEDURE Show C%
510   GLOBAL Pr_Nt, Ln_Fd
520   IF C% = 0 THEN LEAVE
```

 © *RM User Group 1988*

```
530  PUT C% : IF C% = 13 AND Ln_Fd THEN PUT 10
540  IF NOT Pr_Nt THEN LEAVE
550  PUT #2, C% : IF C% = 13 AND Ln_Fd THEN PUT #2, 10
560 ENDPROC
570 :
580 PROCEDURE Look_For Lk_Fr$
590  In$ := ""
600  REPEAT
610    Send Keyb() : C% := Chr_In() : Show C%
620    IF C% > 31 AND C% < 127 THEN In$ := In$ + CHR$(C%)
630  UNTIL RIGHT$(In$, LEN(Lk_Fr$)) == Lk_Fr$
640 ENDPROC
650 :
660 PROCEDURE Download
670  GLOBAL Pg_Dn
680  INPUT "Filename "; File$ : IF File$ = "" THEN 760
690  IF LOOKUP(File$) THEN PRINT "*ALREADY EXISTS*" : GOTO
760
700  CREATE #30, File$
710  REPEAT
720    C% := Chr_In() : IF C% > 0 THEN Show C% : PUT #30,
C%
730    Send Keyb()
740  UNTIL NOT Pg_Dn
750  CLOSE #30
760  NOTE 148, 100, 10 : Pg_Dn := FALSE
770 ENDPROC
780 :
790 PROCEDURE Upload
800  GLOBAL Pg_Up
810  INPUT "Filename "; File$ : IF File$ = "" THEN 880
820  IF NOT LOOKUP(File$) THEN PRINT "*NOT FOUND*" : GOTO
880
830  OPEN #30, File$ : ON EOF #30 Pg_Up := FALSE : CONTINUE
840  REPEAT
850    INPUT LINE #30, Z$ : Send_Str Z$ : C% := Keyb()
860  UNTIL NOT Pg_Up
870  CLOSE #30
880  NOTE 148, 100, 10 : Pg_Up := FALSE
890 ENDPROC
900 :
910 PROCEDURE Send_Str C$
920  FOR C% := 1 TO LEN(C$)
930    Send ASC(MID$(C$, C%, 1)) : Show Chr_In()
940  NEXT C%
```

```
950  Send 13
960 ENDPROC
970 :
980 PROCEDURE Re_Define_Keys
990  PUT 27, "[0;79;255;p" : REM END=END
1000  PUT 27, "[0;81;253;p" : REM PG_DN
1010  PUT 27, "[0;73;254;p" : REM PG_UP=Upload
1020  PUT 27, "[0;71;252;p" : REM HOME=Printer on/off
1030 ENDPROC
```

Cables:

The connector (Bell 600 series) used to bring the Nimbus-to-modem cable into the AUX socket does not identify the individual wires. The numbers given below refer to the sequence as obtained by holding the connector as though to insert it in the AUX socket, then, viewed from above and reading left to right, the connectors can be referred to as:

6 3 5 2 4 1

Nimbus AUX	to	Tandata 512 (25 way D)
Pin 6 - N/C		
Pin 3 - (GND)		Pin 7, Gnd
Pin 5 - Txd		Pin 2, Txd
Pin 2 - Rxd		Pin 3, Rxd
Pin 4 - RTS		Pin 4, RTS
Pin 1 - Gnd		Pin 7, Gnd

 © RM User Group 1988

480Z Quest to dBase II File Transfer

by Richard A. Naylor

This article describes a file transfer program which allows QUEST files to be converted for use under DBASE II. The program has been written in SBAS. Although rather slow in execution this is not really a problem as most QUEST files seem to be quite small and in any case one would normally only do the process once on a file. The use of TURBO Pascal would bring about a marked speed improvement.

Data Structures:
QUEST

QUEST files are basically text files in which fields are of variable length and each occupying one line. The file begins with a header giving details of the structure of the file which is followed by the data.

Header Block:

Number of records
Number of fields
Maximum field length
For each field: Fieldname
Field length
Field Description (for use in EXPLAIN)

Data Block:

For each record: One field per line
(No distinction between types of data all stored in ASCII)

dBASE II

dBASE II uses fixed field lengths for data storage and distinguishes between Numeric, Character and Logical data. The file begins with a 521 byte header block followed by the data.

Header Block:

dBASE ID	One byte contains 02Hex
Number of records	16 bit integer stored low byte first
Date	3 bytes for day,month,year (8 bit integer)
Record length	16 bit integer, low byte first
For each field:	Field name padded to 10 characters with spaces
	Zero byte

 © RM User Group 1988

Field type: 'C','N','L' - one byte ASCII
Field length: 8 bit integer
Zero byte
Zero byte
Zero byte

After the last field there is a carriage return 0D Hex. Remainder of 521 bytes are 0 Hex. This can be calculated using: *bytes=521-9-(n*16)* where *n* is the number of fields.

Data Block:

Each record starts with a space to indicate active or a '*' to indicate deleted. Character fields are stored as ASCII padded to correct field length with spaces.

The Program

In converting the QUEST files the program assumes that all fields are of type Character. This simplifies the conversion process. The dBASE field lengths are simply the lengths of the longest pieces of data in the corresponding QUEST field. The basic algorithm is :

Read QUEST header
Write dBASE header

For each record:

Read QUEST record
Write dBASE record

After converting a QUEST file the dBASE file would normally be modified in the standard way (MODIFY STRUCTURE) to change appropriate fields from Character to Numeric and to establish more suitable field lengths.

Example using PTABLE.QST

Use program below to convert PTABLE.QST to PTABLE.DBF.
Load up dBASE II then:-

```
USE PTABLE
COPY STRUCTURE TO NEWPTABL
USE NEWPTABL
MODIFY STRUCTURE - make required alterations to structure.
APPEND FROM PTABLE
USE
DELETE FILE PTABLE
RENAME NEWPTABL TO PTABLE
```

 © RM User Group 1988

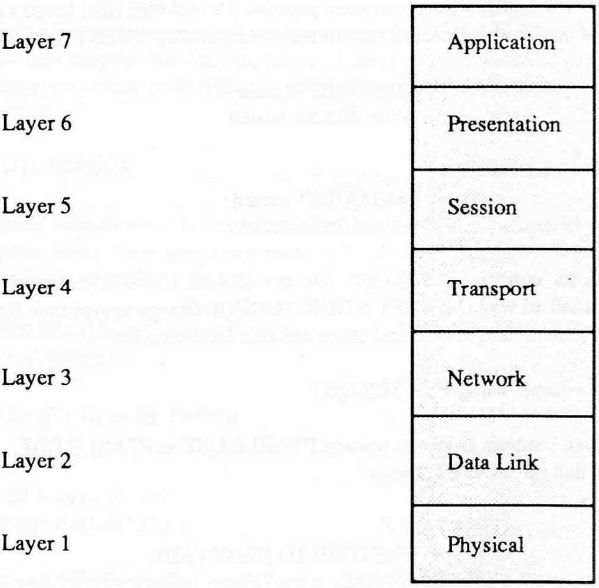
ADVANCED NETWORKING I

by Ray Fleming

ADVANCED NETWORKING

Any look at advanced networking systems must start by looking at the International Standards Organisation (ISO) Open Systems Interconnection (OSI) Reference Model. This is a recognised framework for international networking standards which can be applied to Local Area Networks. The key point within OSI is that systems adhering to this model can communicate openly with other equally compatible systems.

The OSI reference model specifies a series of seven layers for communication, illustrated below:



Layer 1, the Physical Layer involves the actual transmission of data and the physical media used for this. In the case of RM Net this is co-axial cable, but there are a variety of others including fibre optics, twisted pair cable and shielded multicore.

 © RM User Group 1988

Layer 2, the Data Link Layer, involves the techniques of transmission, the physical bit patterns, tokens and error detection. This has resulted in the evolution of the token ring and CSMA/CD protocols (CSMA/CD is used on the Nimbus network: it is a shortened version of Carrier Sensed Multiple Access with Collision Detection).

Layer 3, the Network Layer, involves packet addressing and delivery. It deals with the routing of information between network nodes and interconnecting networks.

Layer 4, The Transport Layer, provides for the delivery of packets and their arrival in the correct sequence.

Layer 5, the Session Layer, is concerned with administrating communications - the sending and receiving of messages.

Layer 6. the Presentation Layer, converts data formats from those suitable for the network, to those suitable for the application.

Layer 7, the Application Layer, provides the interface between the application programs and the network.

In the ISO Model, each layer provides a specific service. The model is hierarchical, meaning that layers connect only with layers immediately above and below. It was created as a way to provide a standard communications interface, for different types of hardware to attach to the network and different applications to run on the network. The model will also allow different networks to communicate with each other.

Let us look at the OSI Model in relation to networks for RM.

 © RM User Group 1988

OSI MODEL	RM NET (Microsoft Networks)	CHAIN NETWORK (DR Net)
7: Application	MS-DOS 3.1	CP/M
6: Presentation	REDIR	CP/NET
5: Session	NETBIOS	
4: Transport		NIOS
3: Network	XPORT	
2: Data Link	ZEP/ZEND	ZEP/ZEND
1: Physical	ZNET	ZNET

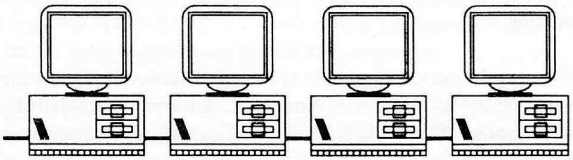
There are only two parts of the OSI Model which have been implemented and adopted by most LAN manufacturers/vendors. These are Layers 5 and 6; the Session and Application layers.

Standardisation of the Application layer is based on Microsoft's MS-DOS 3.1. MS-DOS 3.1 included networking primitives within its standard operating system facilities, to allow multiuser software applications to access the network and to use file and record locking.

The Session layer is standardised through the NETBIOS system. This was adopted by IBM as part of its PC Network. NETBIOS is a software interface for applications which do not want to use the MS-DOS interface. It allows direct station-to-station and station-to-server communication.

NETBIOS does not just cover the Session layer, but also includes the Transport and Network layers. It is important because, in theory, it allows any LAN hardware system that supports NETBIOS to support any NETBIOS-compatible network operating system or hardware device.

Let us take a step backwards and look at a typical network, running RM Net II. The system is based around a single length of co-axial cable, connected to a BNC connector on the rear of the machine. The BNC connector connects the cabling system to the Network Interface Card (NIC). These two combine to form the OSI Model Layer 1, the Physical layer.



The stations and server run the Microsoft Networks operating system (MS NET). As a whole this covers Layers 2 to 7, but this can be broken down further. The Nimbus specific firmware which talks between the operating system and the NIC comprises part of the Data Link Layer. This actually places messages onto the cable, using CSMA/CD principles. It is called ZEP and ZEND.

Continuing upwards, Microsoft Networks provides a layer which spans from the Data Link Layer, through the Network layer and up to the Transport layer. It provides the communication between the layers above and below and is called XPORT.

Carrying on up the hierarchy, NETBIOS occupies the Session layer, as we have already noted. The final two layers are involved with MS-DOS, the Redirector providing for conversion between MS-DOS data and the correct format for network communication.

Finally, layer 7, the Application layer is occupied by the MS-DOS operating system. As you may be aware MS-DOS provides facilities to applications programs to open and close files and to read and write data whether this happens over a network or on a local floppy disk is invisible to the user. There are some extra facilities, provided specifically for network use; for example, the ability to read files at the same time as other users, and exclusively update parts by locking parts of files from other users (record locking).

Because of NETBIOS' facilities, it is possible for workstations on a network to talk directly to each other. This could be used for simple electronic mail systems, or it could be used for more advanced functions such as its use to provide network communications to outside facilities, such as other networks or mainframe computers.

We will look at network communications shortly but first let us examine different networking systems and performance.

ADVANCED NETWORKING II

DIFFERENT NETWORKING SYSTEMS

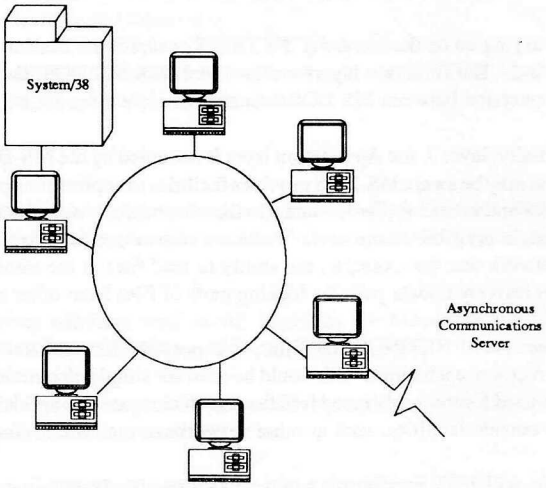
There are two major types of local area networks in use within the UK educational and business establishments. They are based on either Ethernet principles or IBM's Token Ring.

IBM Token Ring

The Token Ring, as the name suggests, is a ring based network system which uses a token passing system for network communication. It was announced originally by IBM as a way of connecting together PC and PC AT micros. The original announcement, in October 1985, allowed for a single ring with a limited distance and a maximum of 260 stations.

In April 1986, a subsequent announcement expanded the options to allow bridged networks, longer distances and connections to IBM's System/36 and System/370.

By this stage the network could allow PC's and compatibles to link together in a local area network with communications to IBM's minis and also to the outside world (using the Asynchronous Communications Server).



 © RM User Group 1988

Later developments in Token Ring have resulted in the Token Ring Network Interface Card becoming available from other sources and network operating systems (other than IBM's) to support the Token Ring system.

Although Token Ring offers many advantages such as the ability to connect to IBM's minis and the conformance to ISO standards, it also has a number of drawbacks, one being its relatively high cost.

For this reason many people use alternative networking systems, many based on Ethernet.

Ethernet Based Networks

Ethernet based networks use a bus cabling system, with a CSMA/CD access technique. CSMA/CD (Carrier Sensed Multiple Access with Collision Detection) allows stations to talk across the cable at any time and detect whether their messages have been corrupted by other workstations talking.

As well as "real" Ethernet, there are many derivatives from the Ethernet standard, many offering similar performance with cheaper cabling costs. Two systems are Z-Net and Cheapernet. The Z-Net system is that used on the Nimbus Network. Although these systems are based on Ethernet principles, they are not able to communicate directly with Ethernet systems.

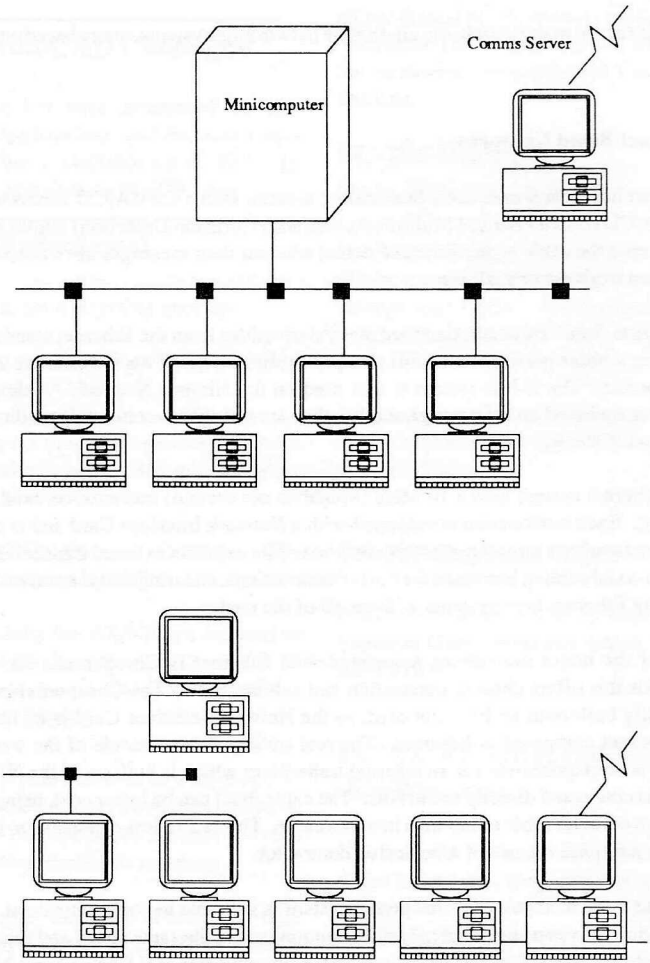
The Ethernet system uses a 10 Mb/s (Megabits per second) bus with co-axial, low loss cabling. Each workstation is equipped with a Network Interface Card and is connected to the network via an external transceiver box. The use of an external transceiver and low loss co-axial cabling increases the cost of connections, ensuring that there are advantages to using Ethernet type systems without all of the costs.

One of the major derivatives associated with Ethernet is Cheapernet. As the name suggests this offers cheaper connection and cabling costs. The Cheapernet interface is normally built onto an Ethernet card, so the Network Interface Card is no cheaper for Cheapernet compared to Ethernet. The real savings occur outside of the workstation. There is no requirement for an external transceiver which is built on to the NIC and the cable is connected directly to the NIC. The cable itself can be lower cost, using standard 50 Ohm co-axial cable rather than low loss cable. This is also less expensive to install and may not require the use of a dedicated contractor.

Despite these changes, the Cheapernet system is still able to communicate at 10_Mb/s. Other derivatives of Ethernet may not communicate at the same speed and this normally reduces the cost of the NIC. Z-Net, which communicates at 0.8 Mb/s has a NIC which costs less than one-third of the cost of a traditional Ethernet card.

As Ethernet is a worldwide standard it can be used by a whole series of different manufacturers and it is possible to use Ethernet to communicate between micros, minicomputers and mainframes.

An Ethernet based network may also cover a large distance with cables capable of running up to almost any distance through the use of fibre optic cable and repeaters.



A typical CheaperNet System

 © RM User Group 1988

NETWORK OPERATING SYSTEMS

It is not just the network hardware which dictates the facilities of the network (eg which equipment can be connected) but also the network operating system.

Like the network hardware, there is also a wide choice of network operating systems. the two "standards" are Microsoft Networks and Novell Netware.

Microsoft Networks have been used by a number of manufacturers to implement LAN's based on many different hardware systems. It is closely associated with Microsoft's MS-DOS, ensuring that it offers the standard file and record locking facilities of MS-DOS version 3.1. The most recent version of Microsoft Networks also provides support for the IBM NETBIOS standard. The Microsoft Networks system does not provide in-built facilities for network bridging or station-to-station communication, but this can be achieved through the use of third party software add-ons. Microsoft Networks, like Novell Netware relies on a dedicated file server which cannot also be run as a workstation. Not all networks, however operate like this and some allow workstations to act as file and print servers while operating as stations. While this may be satisfactory for smaller networks there would be an impact on performance for medium and large network systems.

Novell Netware is available as a number of options, according to which server is being used at the centre of the network, e.g. there are versions specifically for 80286 based servers. The network hardware can be one of a range and there are drivers for 14 Network Interface Cards, supporting 35 LAN configurations. Advanced Netware provides facilities for multiple servers and incorporates software bridges for interconnecting different LAN configurations.

More than one Network Interface Card can be fitted inside a network server, allowing it to talk to two physically different network systems. One could potentially be based on Ethernet, while the other could be based on Token Ring. Stations could talk to any other station, regardless of the network hardware system to which it is connected.

 © RM User Group 1988

ADVANCED NETWORKING III

NETWORK PERFORMANCE

There is one major concern of network users - the performance of their network. The problem is normally caused by a single component within the network. A local area network is a staged system. Data goes through many processes and media, from the initiation of a request at a PC to the reception of the response from the server. Each of these stages is a potential performance bottleneck.

In a network, the primary interaction is between the workstations and a file server - a many-to-one relationship. This brings the server into the position of being a possible candidate for a bottleneck. This does not mean it is always the bottleneck, but it is usually where bottlenecks will occur.

Let us first examine the process of a typical workstation task, for example, a read of a file from the network server:

1. The workstation wishes to read a file. It assembles a network message and passes it to the Network Interface Card.

   Performance is affected by processor and bus speed.

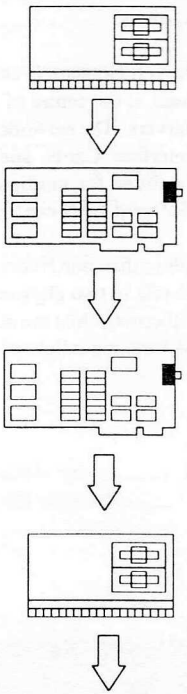
2. The NIC transmits the message to the server, via the network cable.

   Performance is affected by speed of the network (bandwidth) transmission and other network traffic.

3. The server receives the message from its NIC.

   Performance is affected by server processor and bus speed.

4. The server processes the message.



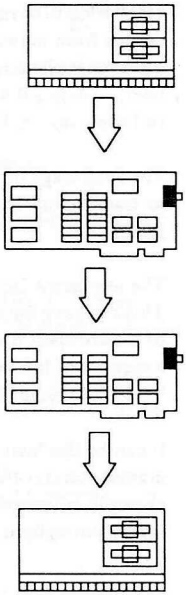
 © RM User Group 1988

5. The server reads the data required from disk. Performance is affected by disk speed and data transfer rate.

6. The message is sent to the NIC.

7. The NIC transfers the message from the server to the station's NIC.

8. The station's NIC transfers the message to the station's main memory.



As you can see, there are a number of processes involved in a simple file transfer. Each process can be affected by different factors which can influence overall network performance. These factors vary from simple processor speed, to network cable bandwidth, to hard disk performance.

Let us start with the server - the most likely place to find a bottleneck. The hard disk speed governs how fast data could possible by read from disk. With some machines that could be as low as 30 to 40 Kbytes/sec. More typically performance is around 64 Kbytes/sec. By using higher speed Winchesters some systems can offer higher data transfer rates. The Nimbus PC based servers offer transfer speeds around 64 Kbytes/sec from hard disk.

The next step is to transfer the data to the NIC. This is normally fast enough not to cause a bottleneck, but there are various techniques for data transfer which have minor affects on this speed. With IBM architecture micros the different choices are the use of Direct Memory Access, IO Mapping and Shared Memory. DMA, the most common form of the three is also the slowest. Even with the slowest form of DMA it is possible to transfer 250 Kb/sec, while faster forms of DMA could give up to 2 Mb/sec.

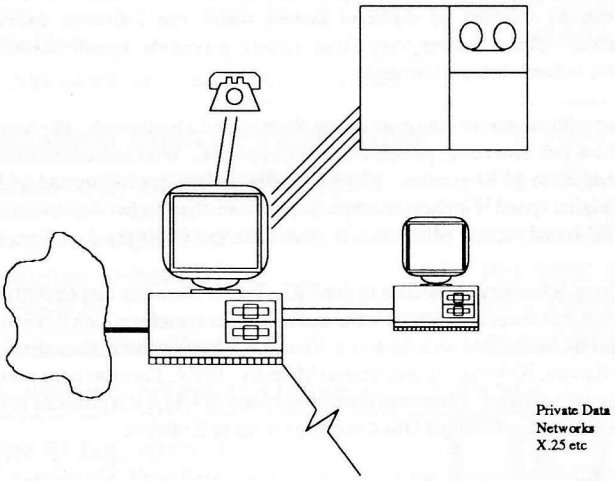
The next transfer is from the NIC in the server to the NIC in the station. The speed of the network transmission affects this transfer, as well as the efficiency of cable usage. An an example, a system which transmits at 0.8 Mb/s results in a maximum capacity of 100 Kb/sec. This assumes that 100% of the cable's capacity is used (very rarely true). From that

100 Kb has to be removed the amount of information which is added to the data itself. This ranges from network addresses for the packet, to error checking values. By the time a message is placed on the cable, 30% of it may be network information, rather than the data itself. So on a 0.8 Mb/sec network, it may only be possible to transmit 65-70 Kb/sec at full capacity. A 10 Mb/sec network could transfer a maximum of around 850 Kb/sec.

The final stage of transferring information from the stations NIC to memory is the same as transferring from the server's memory to the NIC and is constrained by the same factors.

The important factor which has not yet been discussed is the network operating system. This can have the most profound affect on the network performance (and usually it is one of the cheapest parts to change for the user). The network operating system consists of a number of layers and inefficiencies within the layer or in the communication between layers will result in an overall performance loss over the network.

It can be the lowest performing factor within the network and this explains why many manufacturers offer upgrades to the network operating system. The Nimbus Network, for example, offered double performance with a software upgrade. The upgrade boosted total server throughput from 30 Kb/sec to 60 Kb/sec.



ADVANCED NETWORKING IV

NETWORK COMMUNICATIONS

Networks were for many years used simply as a method of sharing a server to provide shared access to files. It was also used to share other physical resources such as printers and plotters. The next stage, following quickly on, was to develop into the use of the network for a simple form of communications - electronic mail. Before the standardisation on NETBIOS, each network system used its own method of interstation communication so that electronic message systems on one network could not run on another. With the standardisation on NETBIOS, it became possible to write electronic mail systems which would work on any network which supported NETBIOS.

With the advances in users' awareness of networking and what it was possible to do, the next stage was obvious - the provision of communications to "the outside world", for example, to minicomputers and mainframes or on to the public telephone system or private data communications networks. The normal way to do this is through a dedicated or non-dedicated communications server. Although the precise details vary according to the network system, the general principles remain the same. It allows specially programmed applications operating on the network to dial out from the network, or outside applications to dial into the network. The server is a program and protocol specification only - on its own it does nothing. As an example, IBM does not provide a terminal emulation package that runs on a workstation and communicates with the server. Terminal emulation packages must be supplied by third parties.

Depending on the network system in use, the server could provide anything between 2 and 16 asynchronous lines. The technique for communication is by using the NETBIOS facilities to enable peer to peer workstation communication. The NETBIOS system allows applications to set up messages, including network information such as the sending station and the receiving station. These can then be passed to the NETBIOS, which transmits them through the network interface, with NETBIOS checking that the message arrived correctly. Using this technique it is possible to talk to a communications server, send information to be transmitted through its serial interface and receive information which has been received down the line.

There are a number of packages, using NETBIOS which can use an Asynchronous Communications Server. They include simple terminal emulators (eg VT100 and VT52), more advanced terminal emulators (eg Tektronix) and other functions. As an example, it is possible to use a package to dial in remotely to the network and operate just like a local network workstation. Using cards other than a standard serial interface, it is possible to connect to other systems. For example, DCA manufacture a number of add-on cards which allow connection to IBM mainframes and minis. With these cards you would use software supplied by DCA and third parties to drive it.

Micro Mart

For Sale:

Nimbus PC-186, 512k, twin floppies. Mono monitor; 5.25" external floppy. Software: Word, Windows, Paintbrush and Logistix. Offers around £900 to:

David Galloway,
97, Pennyland Drive,
Thurso,
Caithness.
0847 - 65330

For Sale:

Nimbus PC-186, 512k, twin floppies. Colour monitor, 5.25" external floppy, mouse, serial piconet. Software: Quest, Windows, Logo. All manuals and disks. Excellent condition. Cost £2300 new. Offers around £1300. Contact:

Steve Aprea on 0344 778824 after 7pm weekdays or all day Saturdays/Sundays.

Current members can advertise free in Micro Mart

Just send your add with current, valid membership number

 © RM User Group 1988

Executive Committee

Chairman	Peter Bishop	23, Larkfield Road, Richmond, Surrey. TW9 2PG **01-948-4382**
Secretary	Tim Davey-Winter	Computing Teaching Cntre, 59, George Street, Oxford. **0865-278826**
Treasurer	Graham Townsend	3, Loseberry Road, Claygate, Esher, Surrey. **0372-66043**
Membership Secretary	Steve Rochford	Hounslow Manor School, Prince Regent Road, Hounslow, Middlesex. TW3 1NE **01-572-4461**
Librarian	Sheila Maynard	7, Wood Drive, Stevenage, Herts. SG2 8PA **0438-354809**
Editor	Robin Davies	13, Paget Road, Wolverhampton. WV6 0DS **0902-310360** (home) **0902-305958** (work)
Advertising	Keith Ashby	Ashby Farms, The Buss Barn Fairfield Road, Appledore Ashford, Kent. TN26 2DR **023-383-378**
Conference	Les Watson	Tulip Cottage, Church Corner, Frampton-on-Severn, Glos GL2 7EQ **0452-741203**
Paul Maskins	Roger Bracey	

 © RM User Group 1988

Parlez-vous Tick-Tack? *Sprichst Du Pen-Friend?*

A new approach to language learning that exploits the micro-computer in a unique way.

Use building-block sentences to create lively foreign text on screen. Then edit it with *built-in multi-lingual text-editor and finally print it out.

You can run the program in English, or in the *target language, and add your own sentences without limit. *MS-DOS only*

★ Provides unrivalled practice in handling the language.

★ Builds confidence through manipulating text on the screen.

★ Accelerates comprehension and develops creative writing skills.

'Pen-Friend' — specially written for the GCSE exam.

Provides strong motivation for teachers and pupils alike. Broad range of ability and age. Ideal for E-mail. Multilingual 'authoring' package.

'Tick-Tack' — learning a language for use in business.

Widely used in Europe by firms who need to communicate in foreign languages, and in the UK by colleges to provide practical language training for business studies, bilingual secretarial courses, BTech, European studies and IT. TVEI in schools.

Available on IBM PC and compatibles, Nimbus, Apricot, BBC; for French, German, Spanish, Italian, Welsh, English (EFL). Other languages too, including Greek and Russian. Books in Chinese, Japanese and Arabic scripts.

'Tick-Tack Project Books' — practical language courses in business communication, new this year. Ideal for Open Learning.

Developed and produced by Anne and David Sephton *(MA Oxon)*
supported by a dedicated team of experts around the world

*Ask for details of bulk contracts and LEA licences enabling schools to acquire packs at low cost.

Free training for key staff.

Primrose Publishing
(the language specialists)
11 Church Street, Thriplow, CAMBRIDGE
(0763) 82512 (hot line)
(0763) 82640 (emergency)